Text Cube: Flight Report Mining by High Dimensional OLAP

Event Cube Research Group, UIUC

Cindy Xide Lin, Bolin Ding, Feida Zhu, Jiawei Han

ASRS Data Set

ACN	Time: Date	Place: Locale	… …	Events: Anomaly	Report
100002	200501	airport : sfo.airport	… …	aircraft equipment problem : critical; non adherence : far	…WHEN TURNING THE NOSEWHEEL, STEERING FAILED…
100045	200501	atc facility : zob.artcc	… …	non adherence : published inflight encounter : birds	… A BIRD STRIKE IN AN ENG AT 1500 FT…
100131	200502	intersection : lubbi	… …	ground encounters : animal	…ON THE MANIFEST WAS A GREAT DANE DOG...
……	……	……	……	……	……

Each commercial flight record consists of three parts (reference 9):

❑ ACN: the unique identity number;

❑ Structured Attributes: 52 categorical attributes are recorded to describe flight conditions such as Time: Date, Time: Local Time of the Day, Place: Locale, Place: State,……;

❑ Unstructured Attribute: Report is written by the pilot to narrate events happening during the flight .

Motivation, Challenge and Proposal

Motivation

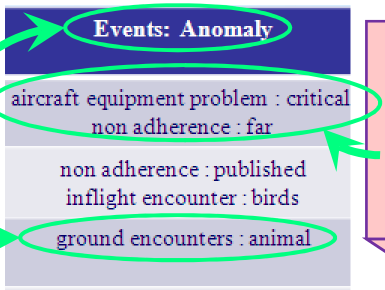
An Organized Approach of Mining Flight Reports for Understanding Anomalous Aviation Events (reference 10)

Challenge

Heterogeneous Data:
- ❑ Structured Categorical Attributes
- ❑ Unstructured Free Text

High Dimensional: the combinations of 52 attributes is exponential.

Concept Hierarchies:
- ❑ Hierarchic Attributes
- ❑ Hierarchic Values



Events: Anomaly
aircraft equipment problem : critical non adherence : far
non adherence : published inflight encounter : birds
ground encounters : animal

Non-deterministic Attribute Values: one record may drop into more than one categories.

Proposal

Text Cube: a novel data cube model that integrates the power of traditional Data Cube (reference 1) and IR techniques (reference 2) for text mining.

Text Cube uses the 52 categorical attributes as Dimensions and the summary statistics on report as Measures.

Targets Challenges

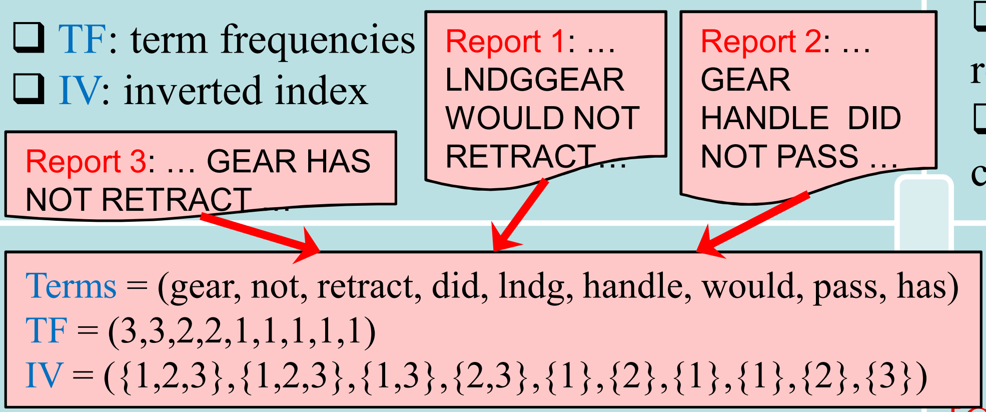
[Challenge 1] Heterogeneous Data

[Solution 1] Text Cube

Dimension: the 52 categorical attributes

Measure: summary statistics on report, including

- ❑ TF: term frequencies
- ❑ IV: inverted index



Report 1: … LNDGGEAR WOULD NOT RETRACT…

Report 2: … GEAR HANDLE DID NOT PASS …

Report 3: … GEAR HAS NOT RETRACT …

Terms = (gear, not, retract, did, lndg, handle, would, pass, has)

TF = (3,3,2,2,1,1,1,1,1)

IV = ({1,2,3},{1,2,3},{1,3},{2,3},{1},{2},{1},{1},{2},{3})

[Challenge 3] Non-deterministic Attribute Value

[Solution 3] Unlike other cubing algorithms such as Multi-Way (reference 5) or BUC (reference 4) which require sorting data due to attribute values, Shell Fragment can put one record into multi cells.

[Challenge 2] High Dimensional

[Solution 2] Shell Fragment (reference 3)

- ❑ Partition the 52 dimensions into several groups (called shell fragments(SF)) ;
- ❑ Compute TF and IV for each SF while retaining fragment inverted index (FIV) ;
- ❑ Given offline-computed SFs, online-compute cube cells by calculating the intersection of FIVs.

[Challenge 4] Concept Hierarchies

[Solution 4] Text Cube has Dimension Hierarchies (reference 8) for dimensions and Term Hierarchies (reference 11) for terms in report. Detail explanation will be given later.

Implement: Preprocessing

Step 1: we utilize WordNet to stem terms (reference 6)

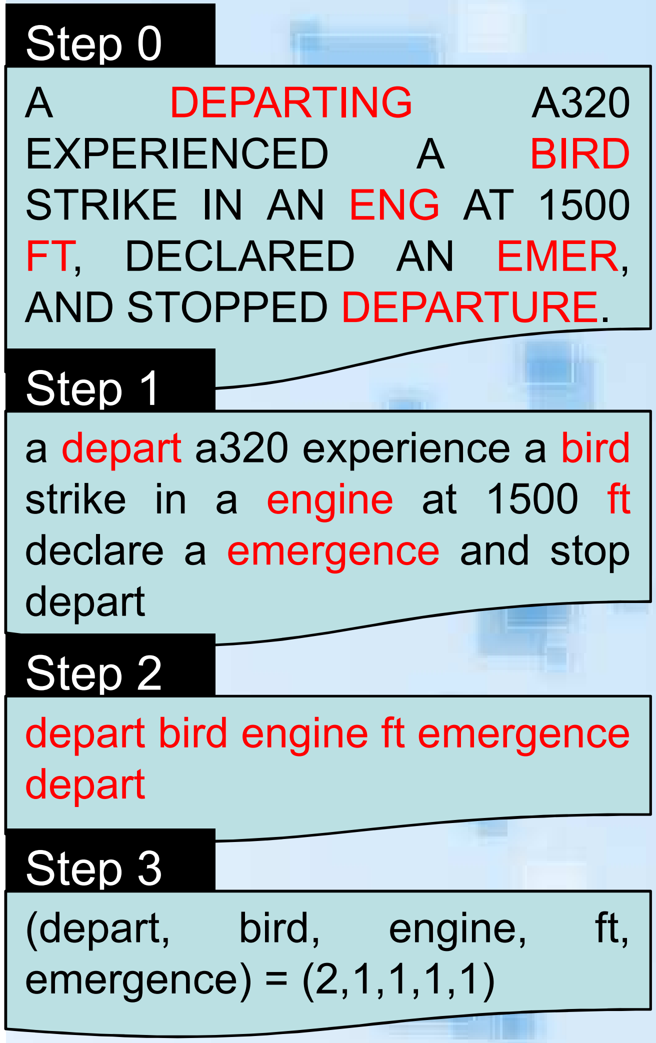
- Before stemming, one word may have different tenses, like steal stole stolen stealing
- After stemming, all words are in the original tense, like steal.

Step 2: not all terms in report need to count in measure.

- Topic Term
  - □ Words that are meaningful for flight records
  - □ Such as altimeter, depart
- Background Term
  - □ Words that are too common and not discriminative
  - □ Such as preposition to, at, and article a, an, the.
- TF-IDF Weighting Formula (reference 7)
  - □ A group of formulas evaluating how important a term is to a document.

we use TF-IDF to weight terms, keep 1000 terms with highest weights as Topic Term, and delete the rest as Background Term.

Step 3: count Term Frequencies.



Step 0

A DEPARTING A320 EXPERIENCED A BIRD STRIKE IN AN ENG AT 1500 FT, DECLARED AN EMER, AND STOPPED DEPARTURE.

Step 1

a depart a320 experience a bird strike in a engine at 1500 ft declare a emergence and stop depart

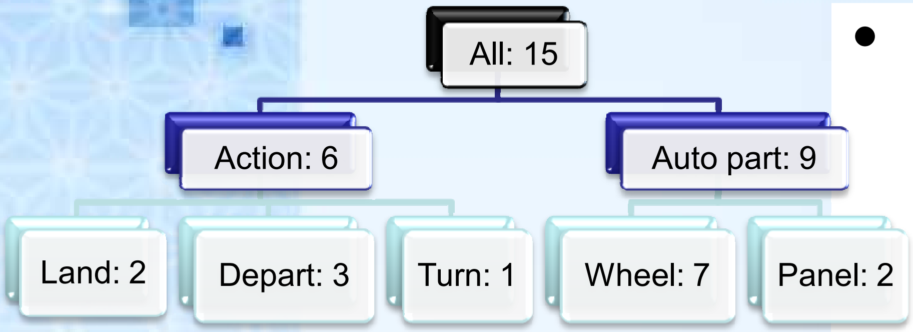
Step 2

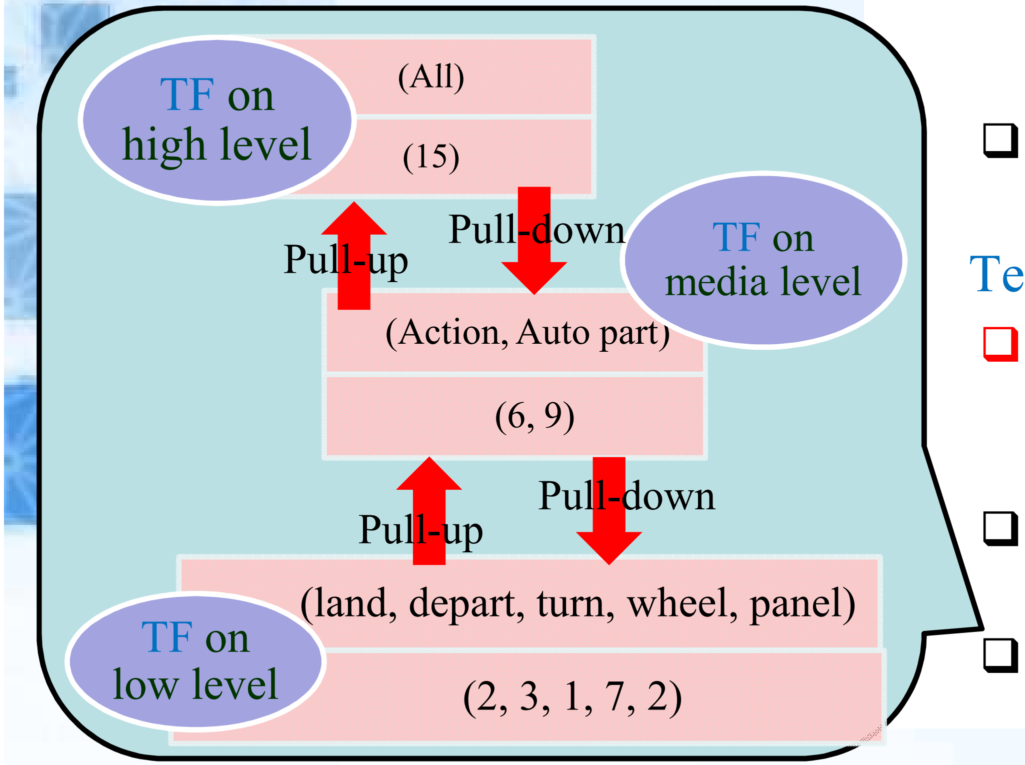
depart bird engine ft emergence depart

Step 3

(depart, bird, engine, ft, emergence) = (2,1,1,1,1)

Implement: Concept Hierarchy





- Dimension Hierarchy (reference 8)
  - □ As traditional OLAP cube, each dimension may consist of more than one attributes, and be organized as a hierarchy of these attributes.
  - □ A dimension hierarchy takes the form of a tree or a DAG. An attribute at a lower level reveals more details.
  - □ Four operations are supported: *roll-up and drill-down, slice and dice*.

Term Hierarchy

- □ Unlike traditional OLAP cube, term hierarchies are novel, which are semantic levels of terms in text report and their relationship.
- □ Term hierarchies are given by aviation experts. It is the way Text Cube introduces expert knowledge.
- □ Two operations are supported: *pull-up* and *pull-down*.

Implement: Complexity

- Time Complexity: (reference 3)

$$O\left(\left\lceil\frac{D}{F}\right\rceil T(2^F-1)K\right)$$

T: # tuples D: # dimensions K: # topic terms F: # dimensions in one shell fragment

- Storage Size:

TF: $O(CK)$ IV: $O(CKT)$

C: # non-empty cells in one shell fragment.

The maximal value of C could be: $O\left(\left\lceil\frac{D}{F}\right\rceil T(2^F-1)\right)$

So the maximal storage sizes of TF and IV could be:

TF: $O\left(\left\lceil\frac{D}{F}\right\rceil T(2^F-1)K\right)$ IV: $O(\left\lceil\frac{D}{F}\right\rceil T^2(2^F-1)K)$

Since K = 1000, D = 52 and F = 1 ~ 4 are all constant, both the time complexity and the storage size of TF are linear to the number of flight records T, but not the storage size of IV.

How to reduce the storage size of IV becomes a new challenge for Text Cube unlike traditional OLAP cube.

Implement: Cubing

- Partially Materialization
  - ❑ To reduce storage size, we select some instead of all cells to materialize
  - ❑ If a non-materialized cell are queried, we online-compute it, based on offline-computed cells.
- Query Time
  - ❑ If a non-materialized cell consists of w materialized sub-cells, we need w times online-computation to answer the query. So the Query Time is defined as w.
- Balance between Space and Time
  - ❑ Given a time threshold Delta, we minimize the storage size while bound the query time of all cells no more than Delta.

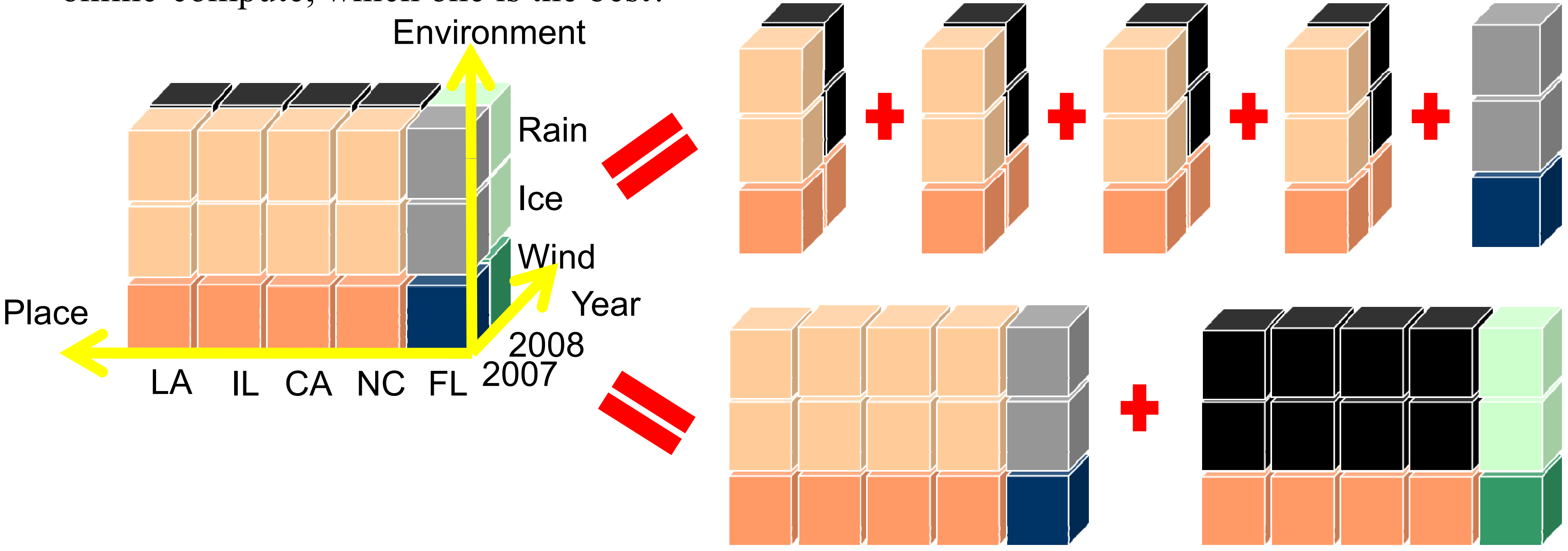
During cubing, we should make decisions on two things:

Q1. Which cells are selected to be materialized ?

Q2. For a non-materialized cell, how to online-compute it ?

Implement: Cubing

- For Q1, we suppose each cell c is not selected and calculate the minimum query time qt(c). If qt(c) > Delta, we materialize c, otherwise record the way to minimize qt(c) for Q2.
- So, Q1 and Q2 both convert into one question: a n-size non-materialized cell have n ways to online-compute; which one is the best?



- For the 3-size cell (*,*,*), the second way seems better then the first one. But actually it depends on the materializing conditions of (LA,*,*), (IL,*,*), (CA,*,*), (NC,*,*), (FL,*,*), (*,2007,*) and (*,2008,*). We use a **Dynamic Programming** model to optimize computation decisions.

Implement: Answer Query

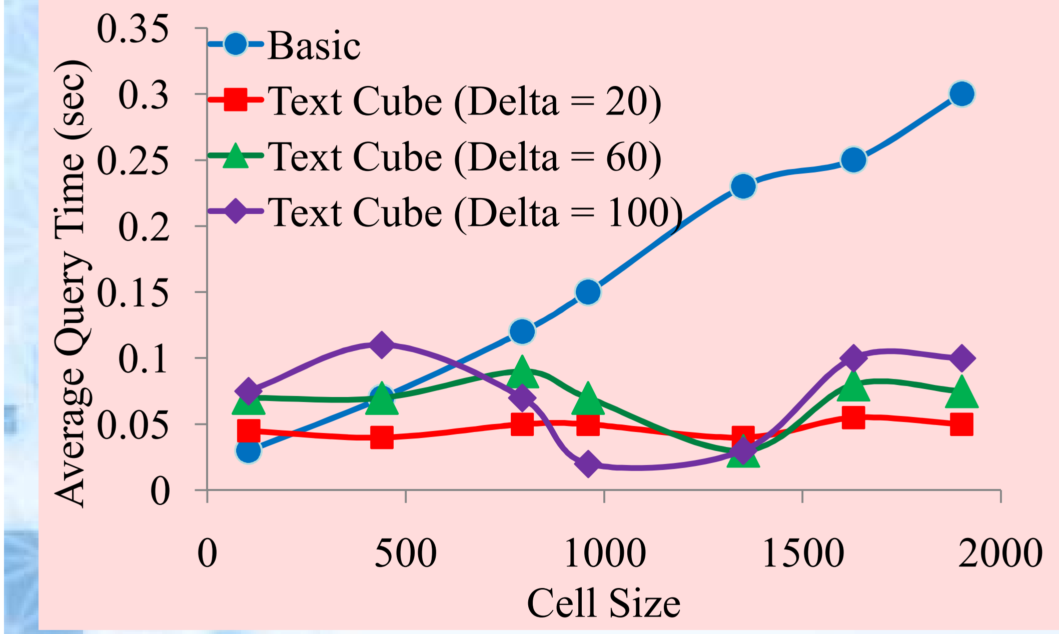
Shell Fragment:

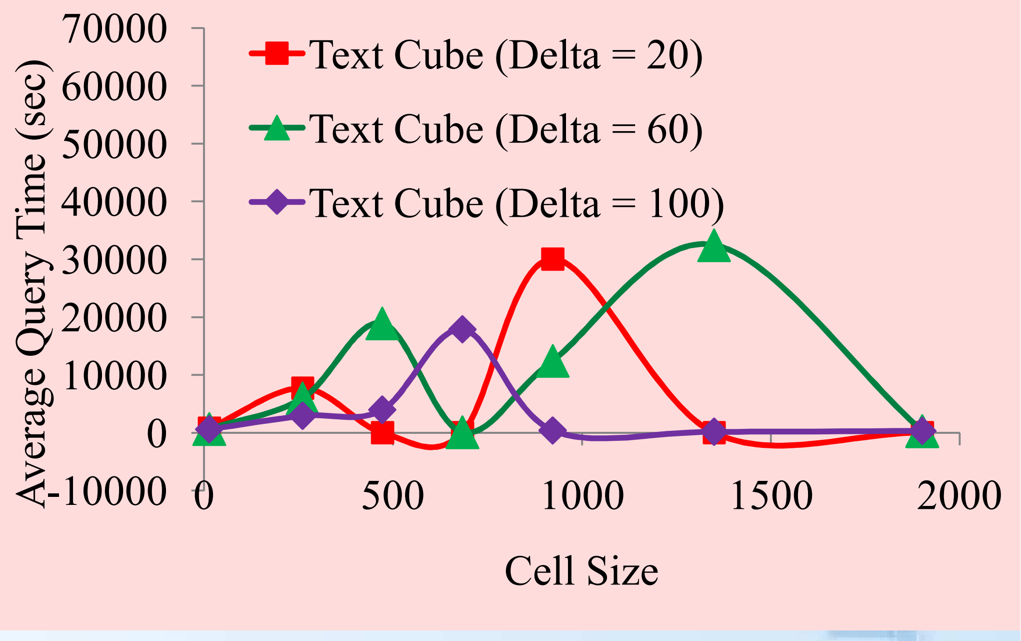
- The 52 attributes are slipped into 16 fragments.
- Regard the 16 shell fragments as 16 text cubes, and partially materialize them.

When online query comes:

- 1. if the queried cells is materialized, we simply output the TF and IV.
- 2. If the queried cell is not materialized but within one fragment, we online-compute w times based on w offline-computed cell.
- 3. If the queried is among several fragments, we intersect their FIV to obtain IV, and go back to database to compute TF.

Experiment: Efficiency

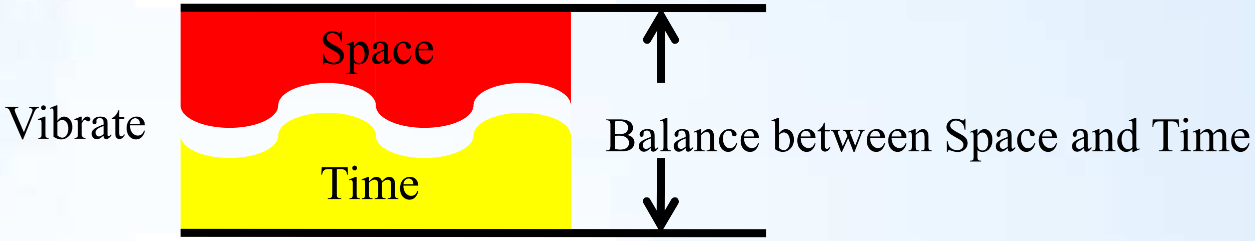




- Cell Size
  - □ The number of aggregated records in a queried cell
- Basic
  - □ A basic algorithm which
    1) retrieve records that match the online query
    2) online-compute TF and IV to answer the query.

Experiment: Efficiency

- Observation and Analysis:
  1. The query time of Basic increases dramatically and linearly as the cell size increases; however, the query times of Text Cube are independent on cell size.
  2. The larger the time threshold Delta is, the dramatically the query time vibrates along the cell size increases;
  3. The query times of Text Cube are always bounded by their time thresholds.
- Reason for the query time vibrating:
  1. At the beginning, all base cells (size = 1) are materialized; then, cells with small sizes are not materialized; further, along the cell size increases, the query times of non-materialized cells are beyond Delta, so cells are materialized again; this happens literately, so the percentage of materialized cells, as well as the storage size, vibrates.
  2. The storage space and the query time are tradeoff. When storage space increases, the query time drops; when the storage space decrease, the query time rises.



Vibrate

Space

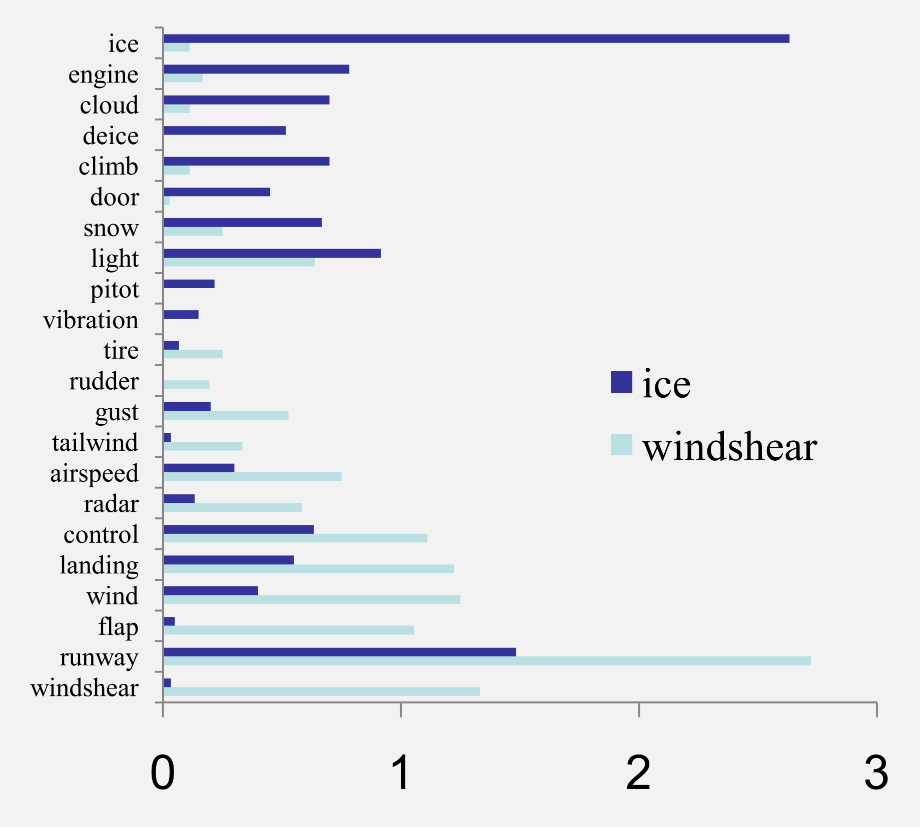
Time

Balance between Space and Time

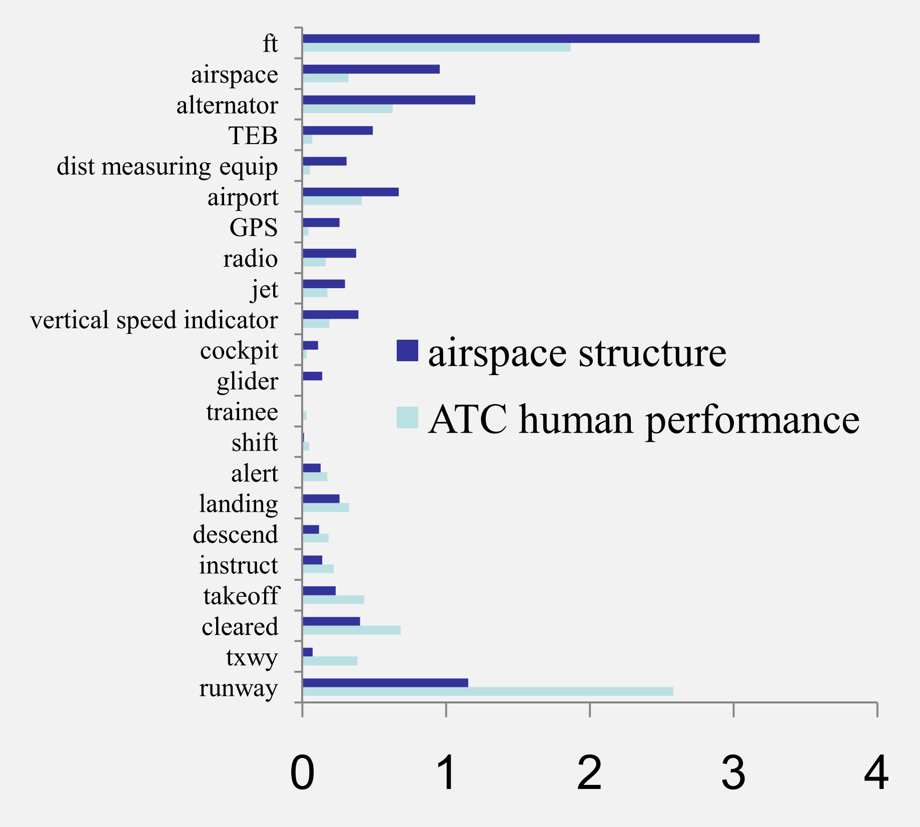
Experiment: Effectiveness

Interesting Result: (avgTF = TF / count)

Compare avgTF under different "*Environment: Weather Elements*"



Compare avgTF under different "*Supplementary: Problem Areas*"



Reference

- 1. Jim Gray, "*Data Cube: A Relational Aggregation Operator Generalizing Group-by, Cross-Tab, and Sub Totals*", Data Mining and Knowledge Discovery (KDD) 1997.
- 2. Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, "*Introduction to Information Retrieval*", Cambridge University Press. 2008.
- 3. Xiaolei Li, Jiawei Han, Hector Gonzalez, "*High-Dimensional OLAP: A Minimal Cubing Approach*", Very Large Database (VLDB), 2004.
- 4. K. S. Beyer and R. Ramakrishnan. "*Bottom-up computation of sparse and iceberg cubes*". In *SIGMOD Conference,* 1999.
- 5. Y. Zhao, P. Deshpande, and J. F. Naughton. "*An array-based algorithm for simultaneous multidimensional aggregates*". In SIGMOD Conference, 1997.
- 6. *wordnet.princeton.edu*
- 7. *denizyuret.com/students/bmizrahi/node30.html*
- 8. V. Harinarayan, A. Rajaraman and J. D. Ullman, "*Implementing data cubes efficiently*", SIGMOD Conference, 1996.
- 9. http://asrs.arc.nasa.gov/
- *10. http://eventcube.atwiki.com/*
- *11.* A. Inokuchi and K. Takeda. "*A method for online analytical processing of text data*". In *CIKM, 2007.*